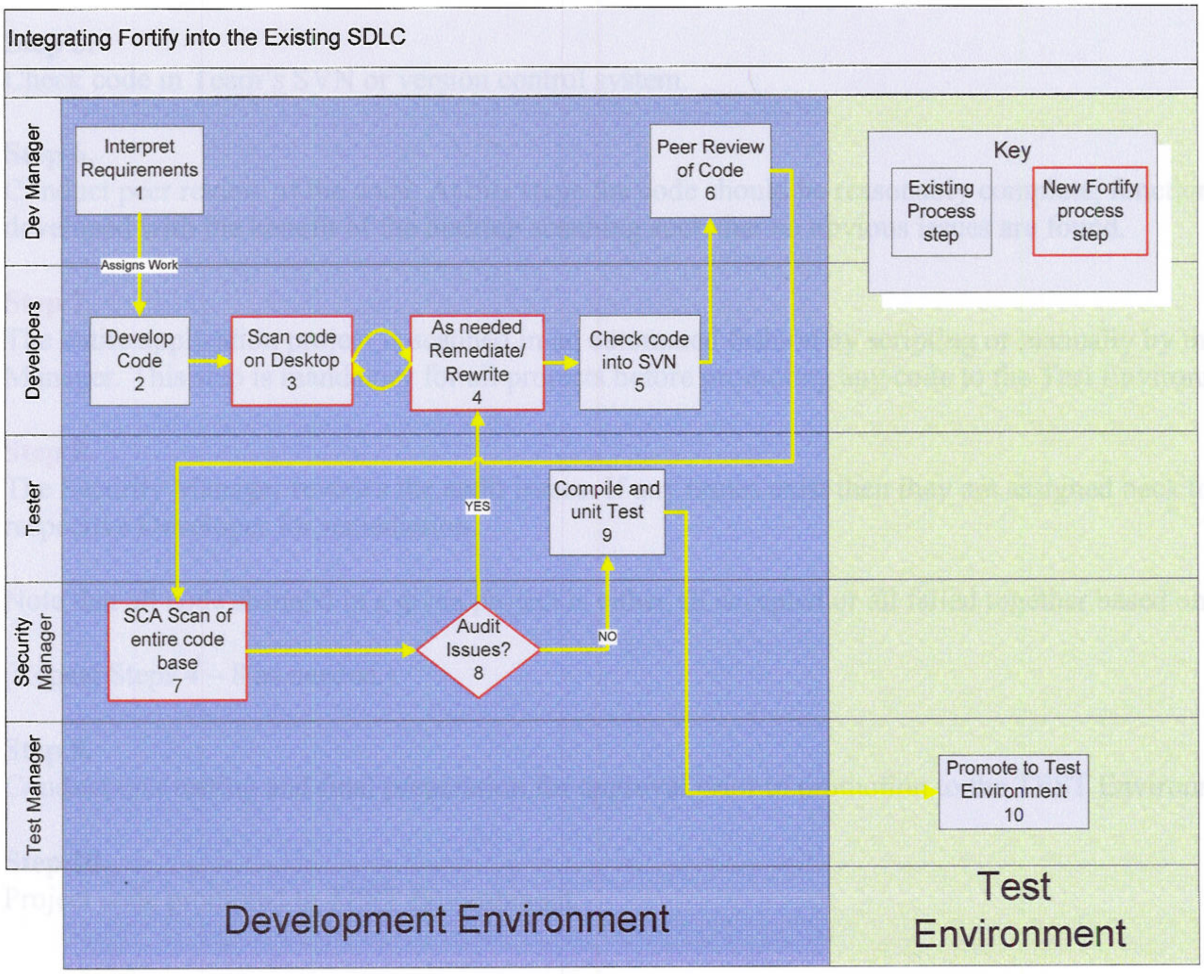


Step 1.
The approved requirements and designs for a new FCR/SCR are received by the Development Team, and broken down into specific assignments.

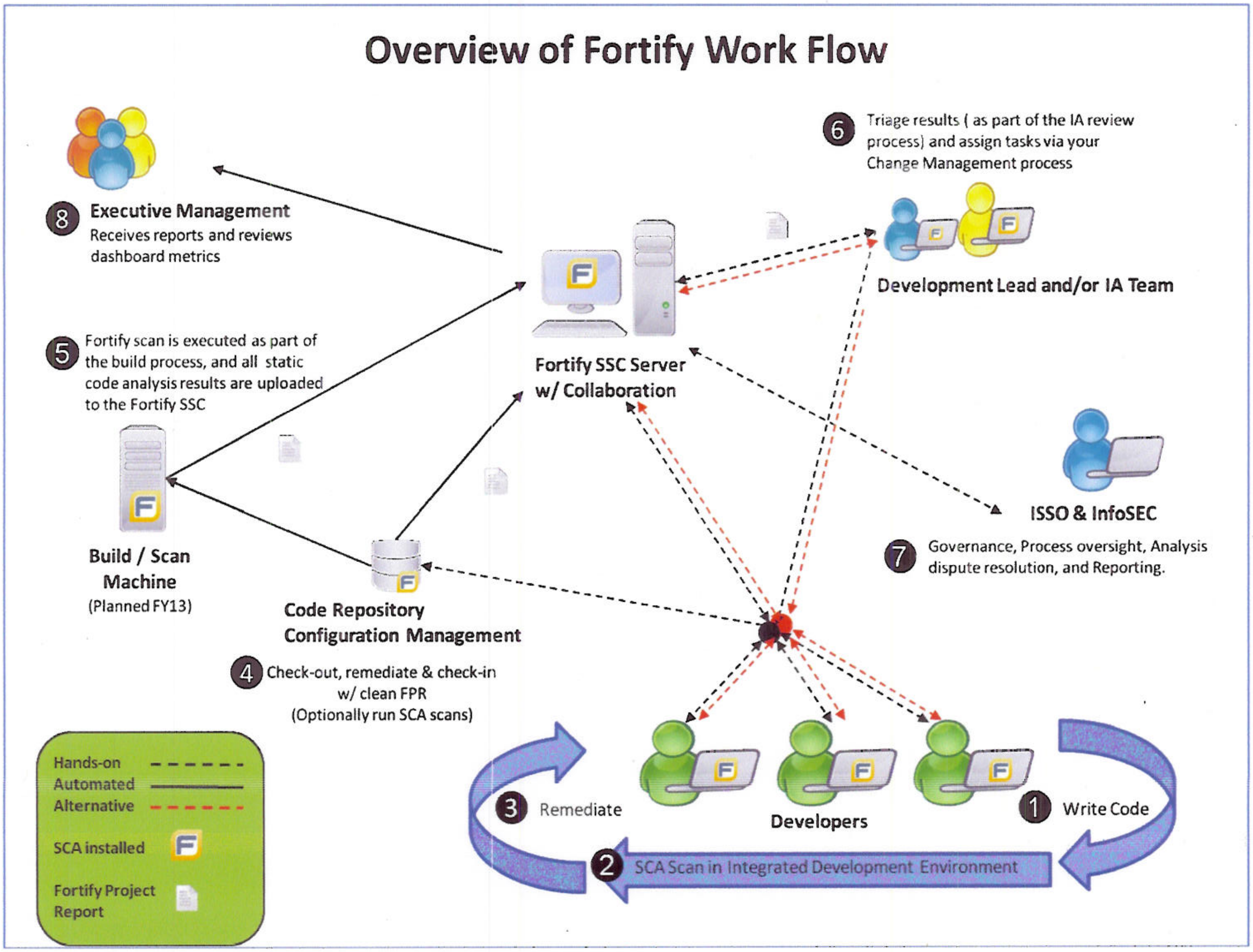
Step 2.
The developers begin to write the code.

Step 3.
Using the Audit workbench tool (aka SCA) developers conduct ad hoc scans of any or all parts of the code they have available to them on their desktop.

Step 4.
Based on the results of the scan, developers review and remediate issues until the code is acceptable for version control system (SVN*) submission. This should be an iterative process by which the developers learn through experience on how to develop better code utilizing the SCA feedback.

*Note: The version control system will be generically referred to as SVN in this document, but any version control system may be used.

(Repeat Steps 3 – 4 as needed.)


Overview of Fortify Work Flow
8 Executive Management
Receives reports and reviews dashboard metrics
6 Triage results (as part of the IA review process) and assign tasks via your Change Management process
Development Lead and/or IA Team
5 Fortify scan is executed as part of the build process, and all static code analysis results are uploaded to the Fortify SSC
Fortify SSC Server w/ Collaboration
ISSO & InfoSEC
7 Governance, Process oversight, Analysis dispute resolution, and Reporting.
Build / Scan Machine
(Planned FY13)
Code Repository Configuration Management
4 Check-out, remediate & check-in w/ clean FPR
(Optionally run SCA scans)
Hands-on
Automated
Alternative
SCA installed
Fortify Project Report
3 Remediate
Developers
1 Write Code
2 SCA Scan in Integrated Development Environment

Step 1. Write code

Developer's checkout and write new code that needs to be tested. The code may be for a completely new project or new code added too, or modified in, an existing project.

Step 2. SCA Scan in Integrated Development Environment

Developers can conduct ad hoc scans against local code in their IDE on demand. Scans conducted in this fashion should be used only for immediate personal code review or for a small working group. These scan reports should not be uploaded on the SSC server.

Step 3. Remediate

Using the information contained in the scan report, the developers should review the issues identified in the new code and work to remediate them before submitting any new code into the version control. Steps 1 -3 may be repeated as many times as necessary to produce error free new code for inclusion in the project.

Step 4. Code Repository Configuration Management

A command line version of the SCA tool maybe installed on the Version Control host for out of band scanning. This process can be automated to submit scans of the complete source code base for final auditing before promoting the code to TEST to the SSC.

Step 5. Build / Scan Machine

In FY13 the project plans to add a dedicated Build/Scan server which can be used for offline scanning. This process can be automated to submit scans of the complete source code base for final auditing before promoting the code to TEST to the SSC.

Step 6. Development Lead and/or IA Team

Scans of the complete source code are reviewed and triaged by a responsible Security Auditor for the project. Issues may be audited and assigned to resources on the team via the SSC console for further review and remediation.

Step 7. ISSO & InfoSec

FAS InfoSec and the ISSO will provide oversight and guidance to the development teams to ensure the Fortify process is setup correctly and functioning. The Divisional/Application ISSO will ensure that all new and existing projects are scanned in accordance with this SOP, and provide day-to-day support and guidance for auditing issues as well. The ISSO will provide immediate feedback on the team's dashboard progress to the Divisional Director and report systemic concerns as needed to InfoSec.

Step 8. Executive Management

Executive Management includes Divisional Directors, System Owners and the Office of the Chief Information Officer. Executive Management will ensure all development projects are properly represented in the SSC and development teams have sufficient resources to address audit issues in a timely fashion. Executive Management will also have read only access to view Dashboard metric and download reports on demand.

2.4. DESKTOP SCANNING

Desktop scanning is performed using the Audit workbench tool suite, sometimes referred to as Static Code Analyzer (SCA) installed in the Integrated Development Environment (IDE). Developers, Security Managers, ISSOs, and FAS InfoSec team members will use SCA to conduct desktop scans and review the resulting FPR files. Desktop scanning is best utilized by hands-on security technicians and developers. All developers are encouraged to use desktop scanning liberally to improve their code skills and decrease the number of coding defects submitted. If someone, such as a Manager, does not normally develop or review source code then they may be better served by accessing the Software Security Center directly. Audit workbench may be installed by submitting a waiver request for non-standard software as outlined in section 2.6.

2.5. AUTOMATED SCANNING

Automated scanning is made possible by way of integrating the fortify SCA scanner on build servers with scripts. FAS InfoSec will assist with the development of these scripts which can save valuable development time and provide a consistent baseline of scan data. Scripted scanning is available on Linux, Windows, or UNIX platforms. Automated scanning has the benefit of performing scans in a routine, consistent manner and also directly uploading the results to the SSC without user intervention. This saves time, and permits the team to remain active with their SCS process with minimal effort. Auditing of automated scans is still required.

2.6. INSTALLATION

2.6.1. Requesting Installation

Fortify software maybe ordered by contacting the IT Service desk to request an automated installation package. It is available for either 32bit or 64 bit windows systems. Local administrator rights are required for manual installation. Detailed installation instructions maybe downloaded from the FAS wiki for a manual installation. (Note: the link is case sensitive)

[redacted]

2.6.2. IDE Plug-ins

Fortify comes with optional Eclipse and Visual Studio plug-ins which are compatible with most versions of Eclipse or MyEclipse. Detailed installation instructions maybe downloaded from the FAS wiki. Note: Due to the variety of Eclipse versions in use, there is no one complete set of instructions to cover all versions. Please follow the directions as a guideline to install the plug-ins in your version of Eclipse. The Eclipse plug-ins may be located on your local computer in these folders depending on your Operating System type:

- C:\Program Files (x86)\Fortify Software\ HP Fortify v3.50\plugins\
- C:\Program Files\Fortify Software\HP Fortify v3.50\plugins\

Note: As of this writing version 3.50 was current. Please be aware that the current version number may be different.

2.7. APPLICATION SCANNING

When configuring a project file within Audit workbench or setting up an SCA script it is a best practice to permit the scanner full access to source code available. Whether or not an audit is being conducted from the desktop or via an automated scan, if the purpose of the scan is to satisfy Step 7 of the Fortify scanning process outlined above, then the entire code base must be included in the project to the greatest extent possible. This means all source code developed by FAS, and all third party libraries and frameworks should also be included. This will improve the overall quality of the audit, and potentially decrease the number of false positives generated by Fortify when it has access to all linked third party sources. If third party frameworks are unavailable, remote, or prohibit scanning or inspection due to licensing constraints, a best effort to include as much code as possible should still be made by addressing the issue. Remote or absent code should be made local to the project file in order to avoid possible network congestion during scans. Third party code which is closed source and not readily available for scanning due to licensing constraints should be considered for use in FAS IT Systems very carefully or discarded in favor of more open sources. Such sources may introduce unknown risks to FAS Applications. Please contact InfoSec for further direction on a case-by-case basis if needed.

If a scan is conducted for any other reason than a complete scan for promotion to TEST or PRODUCTION environments (such as a one-off for a given piece of code only, or a subset of the code from a few developers), then this requirement does not apply. However, attempting to merge the resulting audit FPR file in the SSC console may introduce inaccuracies in the dashboard as well.

3. PROJECT CONFIGURATION

Fortify Projects are the basic unit of sorting and storing SCA scans and audit issues in the SSC. The project file associates the development project and the Fortify scans and the collective history of auditing for tracking purposes and long term storage.

3.1. DEFINING THE PROJECT

Stemming from the kick-off meeting, the development projects which are identified as SSC Projects to be scanned are then defined by a combination of the Division name and the FISMA application name or project name if it is a small application separated by a single dash. All projects shall use the default Audit Workbench Template labeled "AWB Default Template".

By the direction of and the agreement by the System Owner and ISSO, several General Project attributes must be defined before scans are stored in the SSC. The attributes to be defined include the following:

- **Version Information** - Use the same internal version naming scheme already in use for the project.
- **Business Attributes** - The Business Risk shall be the same as the FISMA FIPS 199 Categorization
- **Technical Attributes** - The technical attributes shall describe the application's SDLC phase, accessibility, platform, interfaces, languages used, and other attributes.

3.2. ACCESS CONTROL & ROLES

The SSC is configured to use LDAP authentication. Each Project has a role based access control list associated with it. All accounts are assigned to the prescribed role based on the direction of the Director and ISSO. The defined roles are:

- **Build script** - Used for automated submissions from scripted scans.
- **Developer** - Used to upload analysis results and view audit issues for project versions
- **Manager** - Used to perform the management function on existing Projects
- **Security Lead** - Used to create new Projects and versions
- **View-Only** - Used to review analysis results and reports.

4. AUDITING GUIDANCE

Auditing is the manual process by which a security conscious developer, or analyst, reviews the audit issues found by Fortify and makes a determination what is or isn't valid. This step of the SCS process is best performed by a knowledgeable developer who is familiar with secure coding practices and a working knowledge of the application being audited. When auditing Fortify identified issues, due diligence is required to achieve the highest quality code possible while meeting the competing demands for security, functionality, efficiency, and stability. Due diligence in this context is defined as the process of evaluating each audit finding to determine if it represents a true risk to FAS Information Systems (IS) as outlined by the Fortify audit issue. It is helpful to consider the term *Supplier quality engineering*:

Due diligence is a term used for a number of concepts involving either the performance of source inspection or source surveillance, or the performance of quality duties such as PVA (Process Validation Assessment) or System Audits with a certain standard of care.

Due diligence in supplier quality (also known as due care) is the effort made by an "SQE" (Supplier Quality Engineer) to validate conformance of product provided by the seller to the purchaser. Failure to make this effort may be considered negligence. This is conceptually distinct from investigative due diligence, involving a general obligation to identify true, root cause for non-compliance to meet a standard or contract requirement.

Typically the risk will potentially be manifested in one of three forms against FAS IS or data: Confidentiality, Availability, and Integrity. All three forms are considered to be of equal weight, but the initial risk factor assigned by Fortify must be considered when determining which items must be addressed in order of priority. The analyst must always perform due diligence concerning auditing issues.

4.1. ANALYSIS OPTIONS

4.1.1. Risk Rating

Fortify analysis options rate each audit issue on a scale of risk which is computed from a formula involving the IMPACT of a finding and the LIKELIHOOD it will occur. These values are rated on a 5 point scale in to the following categories as outlined in the Audit workbench User Guide:

- **Critical** - This folder contains issues that have a high impact and a high likelihood of exploitation. You should remediate critical issues immediately.
- **High** - This folder contains issues that have a high impact and a low likelihood of exploitation. You should remediate high issues immediately.
- **Medium** - This folder contains issues that a have low impact and a high likelihood of exploitation. You should remediate Medium issues with the next patch release.
- **Low** - This folder contains issues that have a likelihood of exploitation. You should remediate Low issues with the next patch release.

4.1.2. Analysis State

Each audit issue begins with an analysis setting of **Not Set** indicating that no analysis has been performed and a triage decision is required. The range of Analysis states and their meaning are:

- **Not Set** - This is the default setting of all new issues.
- **Not an Issue** - False Positive finding.
- **Reliability Issue** - This issue could lead to application or system instability. Additional review and remediation is required
- **Bad Practice** - This issue is implemented in potentially functional, but less than optimal, way which might lead to future problems. Review and rework is recommended.
- **Suspicious** - This issue is highly suspect to be exploitable, but no direct proof of the exploit is available. Additional review and remediation is required.
- **Exploitable** - This issue is self-evident or a working exploit has been demonstrated. Remediation is required.

Every issue must be reviewed and a new state set based on the outcome of the analysis. The analysis decision must be made carefully, with due diligence, and completely documented to defend the decision. If the determination is **Not an Issue** or to **Suppress** the finding, then the rationale for that decision must be justified by documenting, in sufficient detail, to make a clear and convincing case as described in Section 4.4.

4.2. SEPARATION OF DUTIES

Auditing activities require a developer to weight competing motivations when charged with auditing their own code since they are managing their own time and meet scheduled deadlines. This situation sets up an inherit conflict of interest. To avoid this situation, a separation of duties is established by assigning the auditing task to another senior developer or team lead as the Security Auditor. No one developer should be responsible for auditing their own code when performing the final scans before promoting to the TEST environment. In cases where a small development group does not have access to a Security Auditor specialist round robin development and auditing maybe used between a pair of coworkers or a small group. It is recommended that each team develop internal standards and guidelines to assist with this duty. Both the ISSO and FAS InfoSec will provide oversight of the process; however it will be incumbent upon the development team to address this risk.

4.3. REQUIRED REMEDIATION ACTIONS

All findings must be reviewed and audited such that all original issues are accounted for by assigning an Analysis state. Note that Suppressing an issue IS NOT a valid audit state. The SUPPRESS function is only useful to remove items from the view; it cannot delete issues from the record. Suppressing an issue may be useful to remove remediated findings from the current view, or items which have been audited as Not an Issue; however they may still show up on reports.

4.3.1. Critical and High

The issues identified as High and Critical vulnerabilities require mitigation before the SCR/FCR is approved and promoted to any testing phase or physical environment of the SDLC. Ideally these items will be removed from the code without an excessive amount of rework and are not the result of a business process requirement.

Note: There is no waiver possible from this requirement.

4.3.2. Medium and Low

The issues identified as Medium and Low should be mitigated as soon as time permits, but within 60 days of the original SCR/FCR being approved and promoted to any testing phase or physical environment of the SDLC. Remediation may still be required at the discretion of the ISSO or ISSM. Ideally these items will be removed from the code without an excessive amount of rework and are not the result of business process requirements.

4.4. REQUIRED DOCUMENTATION

All audited issues require a minimum of documentation in order to support and justify the determination made. If a finding is deemed valid the minimum about of documentation added must be an affirmative statement by the security analyst acknowledging the finding as valid and indicate the SCR or ticket tracking number which is assigned to the finding if available.